Introduction

This module introduces different categories of collections and their operations. It covers the implementation of some particular collections using a concrete data type, called the **array data structure**. The module differentiates between the list and array data structures and walks the students through the **array abstraction architecture** and the operation that can be applied to it, such as capacity resizing, item insertions, and item removal. The extension of the array class will be provided to enable storing and managing rectangular or structured datasets.

Topics and Learning Objectives

Topics

- Overview of collections
- Array and grid data structures
- Operations on arrays and grids
- Introduction to NumPy

Learning Objectives

By the end of the module, you will be able to:

- Implementing collections using the array data structure.
- Insert/remove items from an array.
- Design two-dimensional arrays (grids).
- Use the NumPy library for N-dimensional array operations.

Readings

Required Readings

- Lambert, K. A. (2019). *Fundamentals of Python: Data structures* (2nd ed.). Boston, MA: Cengage Learning.
    - An Overview of Collections, pages 37-48
    - Arrays and Linked Structures, pages 89-102
- McKinney, W. (2017). *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. O'Reilly Media Inc.
    - Read Chapter 4

Introduction to Collections

Please watch this video presentation as it highlights some of the collections that are frequently used in programming. A **collection** is a group of items that can be treated as a single conceptual unit. Although a collection's purpose is to organize data in programs effectively, they differ in structure and use. If you are a developer, you would probably be concerned with how collections can best perform general-purpose resources.

However, if you are a user, you would be concerned with what collections mainly do in various applications. Four types of collections and their operations from the users' perspective will be explored, being **linear**, **hierarchical**, **graph**, and **unordered collections**.

Ryecast Video:
CIND830 - Module 10: Introduction to Collections

Download

Download the PDF file for **CIND830 - Module 10: Arrays and Grids.**

Please note this PDF file will be used in the first three videos of this module. In the PDF, you can use the top navigation bar to navigate between the topics covered in each video.

Download the **CIND830 - Module 10: Introduction to Collections transcript**.

Implementation of Collections

Arrays Data Structures

Please watch this video presentation as it demonstrates how Python is used to implement collections. The main focus will be on **arrays data structures**. The data structure's approach to storing, accessing, and modifying data elements in a particular collection will be explained, highlighting the space-time trade-off that might result from implementing collections using an array class.

Ryecast Video:
CIND830 - Module 10: Arrays Data Structures

Download

Download the PDF file for **CIND830 - Module 10: Arrays and Grids** and click on the topic "**Arrays Data Structures**" in the top navigation bar to take you to the beginning of this topic.

Download the **CIND830 - Module 10: Arrays Data Structures transcript**.

Grids Data Structures

Please watch this video presentation as it shows how to extend the array data structure algorithm to cover two-dimensional arrays, also known as **grids**. For many applications, two-dimensional arrays or grids can be more useful in representing rectangular or structured datasets rather than a single attribute or a column. The presentation below will introduce N-dimensional arrays and ragged grids, and highlight the difference between the latter and the two-dimensional grids.

Ryecast Video:

CIND830 - Module 10: Grids Data Structures

Download

Download the PDF file for **CIND830 - Module 10: Arrays and Grids.** and click on the topic "**Grids Data Structures**" in the top navigation bar to take you to the beginning of this topic.

Download the **CIND830 - Module 10: Grids Data Structures transcript**.

Multidimensional Arrays with the Numpy Package

In this video, a popular scientific computation library called "NumPy" is presented. After introducing the motivation for using this package, we show various multidimensional array operations available in this package.

Ryecast Video:
CIND830 - Module 10: Multidimensional Arrays with the NumPy Package

Download

Download the **CIND830 - Module 10: Multidimensional Arrays with the NumPy Package transcript**.

Self-Assessment Quiz

Please complete the following review activity. The activity is not graded.

Test Yourself

H5P Object:
Self-Assessment Quiz

Summary

In this module, we learned that collections could hold zero or more other objects, and each collection has its operations for accessing its objects, inserting new ones, or traversing them. We learned the architecture of four main collections: linear, hierarchical, graph, and unordered. We explored one of the efficient algorithms to implement a collection, being the array data structure. We saw how arrays perform as random access memory. With no regard to the arrays' size, the interpreter can retrieve any data element in a constant number of steps or in O(1) time. We learned how to insert or remove an item from an array, and how that can be aligned with resizing its capacity. We also extended our concept of arrays to cover the two-dimensional arrays. We ended up our module by introducing the ragged and N-dimensional grids and how they differ from arrays and grids.

Activities

Discussion

Module 10 Discussion

This week, let's discuss arrays and grids in the **Module 10 Discussion** topic in D2L under the **Communication** drop-down menu, then **Discussions**. Feel free to start a new thread or respond to someone else's post.

Lab Activity

Lab 8

Before starting Lab 8, be sure you have reviewed all the content in Module 10. The lab is a practical extension to apply your knowledge gained from this module. Download Lab 8 now and complete it by the end of this week. Your instructor will give you time to work on it before releasing the solutions in a folder called **Lab Solutions** in D2L under the **Course Materials** drop-down menu for you to verify your answers. The lab activity is not graded.

While working on your lab, post any questions, ideas, and strategies in the **Module 10 Discussion** topic in D2L. Feel free to read and respond to your peers' posts as well. You may also have the option of reaching out to your instructor during his/her virtual office hours.

Please note: The PDF for the lab activity may not be fully accessible for students who use assistive technology. If you require an accessible version, please contact your instructor.

Assessments

Quiz Module 10

Please go to your D2L course shell, under the **Assessment** drop-down menu, select **Quizzes**. Then click on the **Quiz Module 10 link** and follow the instructions to complete this quiz. The date and length of time have been set for this quiz. **This quiz is a cumulative 10% of your final grade.**

Reminder

Continue working on **Assignment 2 (worth 15% of your final grade)**, due by the end of Module 12.

While you are working on this assignment, feel free to share your questions, ideas and strategies in the **Assignment 2 Discussion** topic in D2L.

References

- Lambert, K. A. (2019). *Fundamentals of Python: Data structures* (2nd ed.). Boston, MA: Cengage Learning.
- Python Software Foundation. (2020). Python 3.7.8 documentation. Retrieved from https://docs.python.org/3.7/index.html.